

CLAIMS

What is claimed is:

- 1 1. A method for storing data in a repository comprising the steps of:
 - 2 storing in a first table data for one or more default attributes of a first object type used
 - 3 by an application;
 - 4 storing in a second table, separate from said first table, data for one or more default
 - 5 attributes of a second object type used by said application; and
 - 6 storing in a third table, separate from said first and second tables, data for a first
 - 7 custom attribute of said first object type and data for a second custom attribute
 - 8 of said second object type, wherein said first custom attribute and said second
 - 9 custom attribute have the same data type.
- 1 2. The method of Claim 1, wherein:
 - 2 said third table includes
 - 3 at least one instance-identifying column, wherein each row of said third table
 - 4 stores in said at least one instance-identifying column data that
 - 5 uniquely identifies an object instance that is associated with the row;
 - 6 at least one attribute-identifying column, wherein each row of said third table
 - 7 stores in said at least one attribute-identifying column data that
 - 8 identifies a custom attribute of the object instance that is associated
 - 9 with the row; and
 - 10 at least one value column, wherein each row of said third table stores in said at
 - 11 least one value column one or more values for the custom attribute that
 - 12 is identified in said at least one attribute-identifying column; and

13 said at least one value column of said third table stores data that has the same data
14 type as said first custom attribute and said second custom attribute.

1 3. The method of Claim 1, further comprising the step of upgrading said application,
2 wherein upgrading said application comprises the steps of:

3 processing the data stored in said first table, wherein processing comprises:
4 creating a first replacement table to hold the data from said first table;
5 copying the data from said first table to said first replacement table, wherein
6 data from said one or more default attributes of said first object type is
7 copied from said first table into said first replacement table; and
8 deleting said first table;

9 processing the data stored in said second table, wherein processing comprises:

10 creating a second replacement table to hold the data from said second table;
11 copying the data from said second table to said second replacement table,
12 wherein data from said one or more default attributes of said second
13 object type is copied from said second table to said second
14 replacement table; and
15 deleting said second table; and

16 retaining, in said third table, values for said first custom attribute of said first object
17 type and said second custom attribute of said second object type.

1 4. The method of Claim 1, further comprising the step of retrieving an object instance of
2 said first object type, wherein said object instance of said first object type includes data from
3 said third table associated with said first custom attribute of said first object type.

1 5. The method of Claim 1, wherein:

2 said third table stores values for custom attributes of a plurality of object types
3 including said first object type and said second object type; and
4 the method further comprises assigning to every instance of every object type of said
5 plurality of object types an instance identifier value that is unique relative to
6 every instance of every object type of said plurality of object types.

1 6. The method of Claim 1, further comprising the step of storing, in a catalog table, data
2 defining said first custom attribute of said first object type and said second custom attribute
3 of said second object type.

1 7. The method of Claim 6, wherein said catalog table includes:
2 at least one first catalog column, wherein each row of said catalog table stores, within
3 said at least one first catalog column, data that identifies an object type
4 associated with the row,
5 at least one second catalog column, wherein each row of said catalog table stores,
6 within said at least one second catalog column, data that identifies a custom
7 attribute of the object type that is associated with the row, and
8 at least one third catalog column, wherein each row of said catalog table stores,
9 within said at least one third catalog column, data identifying a data type of
10 the custom attribute that is identified in said second catalog column.

1 8. The method of Claim 7, the method further comprising the step of retrieving, in
2 response to a request for an object instance of said first object type, the value of said first
3 custom attribute associated with said object instance by performing the steps of:
4 determining the data type of said first custom attribute from said third catalog column
5 of a catalog-table row stored in said catalog table, wherein:

6 data in said at least one first catalog column of said catalog-table row matches
7 data identifying said first object type; and
8 data in said at least one second catalog column of said catalog-table row
9 matches data identifying said first custom attribute;
10 based on the data type of said first custom attribute, determining the identity of said
11 third table; and
12 retrieving, from a value column of a third-table row stored in said third table, the
13 value of said first custom attribute, wherein:
14 data uniquely identifying said object instance matches data in at least one
15 instance-identifying column of said third-table row; and
16 data identifying said first custom attribute matches data in at least one
17 attribute-identifying column of said third-table row.

1 9. The method of Claim 7, further comprising the step of storing in said catalog table
2 data defining a custom object type, separate from said first object type and said second object
3 type, wherein the step of storing said custom object type includes the step of inserting a row
4 into said catalog table, wherein said row includes:
5 within said at least one first catalog column, data that identifies said custom object
6 type;
7 within said at least one second catalog column, data that identifies a custom attribute
8 of said custom object type, and
9 within said at least one third catalog column, data identifying a data type of said
10 custom attribute that is identified in said at least one second catalog column.

1 10. The method of Claim 6, wherein:

2 said catalog table stores data that identifies custom attributes for a plurality of object
3 types;
4 the method further comprises performing the following steps in response to said
5 application being launched:
6 reading said catalog table to determine custom attributes from said plurality of
7 object types; and
8 based on the information from said catalog table, constructing in volatile
9 memory data structures that indicate the custom attributes of each of
10 said plurality of object types; and
11 in response to a request to access an object instance of a particular object type of said
12 plurality of object types, inspecting said data structures, without accessing
13 said catalog table, to determine the custom attributes of said particular object
14 type.

1 11. The method of Claim 6, wherein:
2 said catalog table stores data that identifies custom attributes for at least one object
3 type;
4 the method further comprises performing the following steps in response to a request
5 to access an object instance of a particular object type:
6 reading said catalog table to determine the custom attributes of said particular
7 object type; and
8 based on the information from said catalog table, constructing in volatile
9 memory data structures that indicate the custom attributes of said
10 particular object type; and

in response to a subsequent request to access an object instance of said particular object type, inspecting said data structures, without accessing said catalog table, to determine the custom attributes of said particular object type.

12. A method for storing data in a repository comprising the steps of:

storing in a first table data for one or more default attributes of an object type used by an application;

storing in a second table, separate from said first table, data for a first custom attribute of said object type that is of a first data type; and

storing in a third table, separate from said first and second tables, data for a second custom attribute of said object type that is of a second data type, wherein said first custom attribute and said second custom attribute have different data types.

13. The method of Claim 12, wherein:
 1. said object type is a first object type of a plurality of object types used by said application; and
 2. the method further includes:
 1. storing in a fourth table, separate from said first, second and third tables, data for one or more default attributes of a second object type of said plurality of object types; and
 2. storing in said second table data for a third custom attribute of said second object type, wherein said third custom attribute of said second object type is of said first data type.

1 14. The method of Claim 12, wherein:

2 said second table includes
3 at least one instance-identifying column, wherein each row of said second
4 table stores in said at least one instance-identifying column data that
5 uniquely identifies an object instance that is associated with the row;
6 at least one attribute-identifying column, wherein each row of said second
7 table stores in said at least one attribute-identifying column data that
8 identifies a custom attribute of the object instance that is associated
9 with the row; and
10 at least one value column, wherein each row of said second table stores in said
11 at least one value column one or more values for the custom attribute
12 that is identified in said at least one attribute-identifying column; and
13 said at least one value column of said second table stores data that has the same data
14 type as said first custom attribute.

1 15. The method of Claim 12, further comprising the step of upgrading said application,
2 wherein upgrading said application comprises the steps of:
3 processing the data stored in said first table, wherein processing comprises:
4 creating a first replacement table to hold the data from said first table;
5 copying the data from said first table to said first replacement table, wherein
6 data from said one or more default attributes of said object type is
7 copied from said first table into said first replacement table; and
8 deleting said first table;
9 retaining, in said second table, values for said first custom attribute of said object
10 type; and

11 retaining, in said third table, values for said second custom attribute of said object
12 type.

1 16. The method of Claim 12, further comprising the step of retrieving an object instance
2 of said object type, wherein said object instance of said object type includes data from said
3 second table associated with said first custom attribute of said object type.

1 17. The method of Claim 12, wherein:
2 said second table stores values for custom attributes of a plurality of object types
3 including said object type, wherein the custom attributes are of said first data
4 type;
5 said third table stores values for custom attributes of a plurality of object types
6 including said object type, wherein the custom attributes are of said second
7 data type; and
8 the method further comprises assigning to every instance of every object type of said
9 plurality of object types an instance identifier value that is unique relative to
10 every instance of every object type of said plurality of object types.

1 18. The method of Claim 12, further comprising the step of storing, in a catalog table,
2 data defining said first custom attribute of said object type and said second custom attribute
3 of said object type.

1 19. The method of Claim 18, wherein said catalog table includes:
2 at least one first catalog column, wherein each row of said catalog table stores, within
3 said at least one first catalog column, data that identifies an object type
4 associated with the row,

5 at least one second catalog column, wherein each row of said catalog table stores,
6 within said at least one second catalog column, data that identifies a custom
7 attribute of the object type that is associated with the row, and
8 at least one third catalog column, wherein each row of said catalog table stores,
9 within said at least one third catalog column, data identifying a data type of
10 the custom attribute that is identified in said second catalog column.

1 20. The method of Claim 19, the method further comprising the step of retrieving, in
2 response to a request for an object instance of said object type, the value of said first custom
3 attribute associated with said object instance by performing the steps of:
4 determining the data type of said first custom attribute from said third catalog column
5 of a catalog-table row stored in said catalog table, wherein:
6 data in said at least one first catalog column of said catalog-table row matches
7 data identifying said object type; and
8 data in said at least one second catalog column of said catalog-table row
9 matches data identifying said first custom attribute;
10 based on the data type of said first custom attribute, determining the identity of said
11 second table; and
12 retrieving, from a value column of a second-table row stored in said second table, the
13 value of said first custom attribute, wherein:
14 data uniquely identifying said object instance matches data in at least one
15 instance-identifying column of said second-table row; and
16 data identifying said first custom attribute matches data in at least one
17 attribute-identifying column of said second-table row.

1 21. The method of Claim 19, further comprising the step of storing in said catalog table
2 data defining a custom object type, separate from said object type, wherein the step of storing
3 said custom object type includes the step of inserting a row into said catalog table, wherein
4 the row includes:

5 within said at least one first catalog column, data that identifies said custom object
6 type;

7 within said at least one second catalog column, data that identifies a custom attribute
8 of said custom object type, and

9 within said at least one third catalog column, data identifying a data type of said
10 custom attribute that is identified in said at least one second catalog column.

1 22. The method of Claim 18, wherein:

2 said catalog table stores data that identifies custom attributes for a plurality of object
3 types;

4 the method further comprises performing the following steps in response to said
5 application being launched:

6 reading said catalog table to determine custom attributes from said plurality of
7 object types; and

8 based on the information from said catalog table, constructing in volatile
9 memory data structures that indicate the custom attributes of each of
10 said plurality of object types; and

11 in response to a request to access an object instance of a particular object type of said
12 plurality of object types, inspecting said data structures, without accessing

13 said catalog table, to determine the custom attributes of said particular object
14 type.

1 23. The method of Claim 18, wherein:
2 said catalog table stores data that identifies custom attributes for at least one object
3 type;
4 the method further comprises performing the following steps in response to a request
5 to access an object instance of a particular object type:
6 reading said catalog table to determine the custom attributes of said particular
7 object type; and
8 based on the information from said catalog table, constructing in volatile
9 memory data structures that indicate the custom attributes of said
10 particular object type; and
11 in response to a subsequent request to access an object instance of said particular
12 object type, inspecting said data structures, without accessing said catalog
13 table, to determine the custom attributes of said particular object type.

1 24. A computer-readable medium, having a database stored therein, said database
2 comprising:
3 a first table storing data for one or more default attributes of a first object type;
4 a second table, separate from said first table, storing data for one or more default
5 attributes of a second object type; and
6 a third table, separate from said first and second tables, storing data for a first custom
7 attribute of said first object type and data for a second custom attribute of said

8 second object type, wherein said first custom attribute and said second custom
9 attribute have the same data type.

1 25. The computer-readable medium of Claim 24, further comprising a fourth table,
2 separate from said first, second and third tables, storing data for a third custom attribute of
3 said first object type, wherein said third custom attribute is of a data type that is different than
4 the data type of said first custom attribute and said second custom attribute.

1 26. The computer-readable medium of Claim 24, wherein:

2 said third table includes

3 at least one instance-identifying column, wherein each row of said third table
4 stores in said at least one instance-identifying column data that
5 uniquely identifies an object instance that is associated with the row;
6 at least one attribute-identifying column, wherein each row of said third table
7 stores in said at least one attribute-identifying column data that
8 identifies a custom attribute of the object instance that is associated
9 with the row; and

10 at least one value column, wherein each row of said third table stores in said at
11 least one value column one or more values for the custom attribute that
12 is identified in said at least one attribute-identifying column; and
13 said at least one value column of said third table stores data that has the same data
14 type as said first custom attribute and said second custom attribute.

1 27. The computer-readable medium of Claim 24, further comprising a catalog table for
2 storing data that defines said first custom attribute of said first object type and said second
3 custom attribute of said second object type.

1 28. The computer-readable medium of Claim 27, wherein said catalog table includes:
2 at least one first catalog column, wherein each row of said catalog table stores, within
3 said at least one first catalog column, data that identifies an object type
4 associated with the row,
5 at least one second catalog column, wherein each row of said catalog table stores,
6 within said at least one second catalog column, data that identifies a custom
7 attribute of the object type that is associated with said row, and
8 at least one third catalog column, wherein each row of said catalog table stores,
9 within said at least one third catalog column, data identifying a data type of
10 the custom attribute that is identified in said second catalog column.

1 29. The computer-readable medium of Claim 28, wherein said catalog table stores data
2 defining a custom object type, separate from said first object type and said second object
3 type, wherein the step of storing said custom object type includes the step of inserting a row
4 into said catalog table, wherein the row includes:
5 within said at least one first catalog column, data that identifies said custom object
6 type;
7 within said at least one second catalog column, data that identifies a custom attribute
8 of said custom object type, and
9 within said at least one third catalog column, data identifying a data type of the
10 custom attribute that is identified in said at least one second catalog column.

1 30. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 1.

1 31. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 2.

1 32. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 3.

1 33. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 4.

1 34. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 5.

1 35. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 6.

1 36. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 7.

1 37. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 8.

1 38. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 9.

1 39. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 10.

1 40. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 11.

1 41. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 12.

1 42. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 13.

1 43. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 14.

1 44. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 15.

1 45. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 16.

1 46. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 17.

1 47. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 18.

1 48. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 19.

1 49. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 20.

1 50. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 21.

1 51. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 22.

1 52. A computer-readable medium carrying one or more sequences of instructions which,
2 when executed by one or more processors, causes the one or more processors to perform the
3 method recited in Claim 23.